

**WEST****Freeform Search****Database:**

US Patents Full-Text Database  
 US Pre-Grant Publication Full-Text Database  
 JPO Abstracts Database  
 EPO Abstracts Database  
 Derwent World Patents Index  
 IBM Technical Disclosure Bulletins

**Term:**

L1 and (packet\$ with header with include\$ and  
 title)

**Display:**  **Documents in Display Format:**  **Starting with Number** 
**Generate:** ☐ Hit List ☒ Hit Count ☐ Side by Side ☐ Image

Search

Clear

Help

Logout

Interrupt

Main Menu

Show \$ Numbers

Edit \$ Numbers

Preferences

Cases

**Search History**
**DATE:** Friday, July 25, 2003   [Printable Copy](#)   [Create Case](#)
**Set Name Query**

side by side

*DB=USPT; PLUR=YES; OP=ADJ*L4   L1 and (packet\$ with header with include\$ and title)L3   L1 and (packet\$ and header and include\$ and title).ab.L2   L1 and (packet\$ and header and include\$).ab.L1   ((709/\$)!.CCLS.)**Hit Count Set Name**

result set

42   L40   L361   L216339   L1

END OF SEARCH HISTORY

## WEST



Generate Collection

L4: Entry 18 of 42

File: USPT

Oct 16, 2001

DOCUMENT-IDENTIFIER: US 6304914 B1

**\*\* See image for Certificate of Correction \*\***

TITLE: Method and apparatus for pre-compression packaging

Detailed Description Text (22):

In data server 200, data server applications 202 generate data packets 204 that are provided to a pre-packager 206. Such data packets can include, for example, news, sports, stock and weather updates. Preferably, each of the data packets includes a header that has an address describing the particular mobile device that the data packet is directed toward. In some embodiments of the invention, the header also includes a sub-address, referred to as a group by the present inventors, that provides a description of a particular channel in the mobile device to which the data is assigned.

Detailed Description Text (23):

Pre-packager 206 stores the data packets for a period of time discussed further below, and after that period of time merges together data packets that have the same address and in some embodiments the same sub-address to form merged data packets. Each of the merged data packets preferably includes a header that describes the contents and destination for the merged data packet. Under these embodiments, for each address and sub-address, pre-packager 206 provides a merged data packet 208 to a router 209 along with the address and sub-address associated with the merged data packet.

Detailed Description Text (25):

Compressed data packets 212 are provided to router 209, which appends a routing tag list to each compressed data packet to indicate the types of translations that were performed on the data packet and to indicate that the data packet is a merged data packet that has a plurality of smaller data packets each destined for the same address and under one embodiment the same sub-address. Router 209 then provides this combined packet to transport module 214, which adds a master header to the combined packet. The master header preferably includes the address and sub-address for the combined packet to form transport data 216, which is provided to transmitter 218. Transport 214 can also reformat the combined data packet it receives so that transport data 216 meets certain restrictions of the communications channel over which transmitter 218 transmits transport data 216.

Detailed Description Text (30):

Once all of the necessary translations have been performed on the data to produce a recovered merged data packet, router 228 routes the recovered merged packets to an exploder 236 based on a tag in the routing list. In one embodiment, exploder 236 examines each of the title IDs in the header of the merged data packet to determine if mobile device 18 has been configured to filter any of the title IDs. For each title ID that is filtered, exploder 236 ignores the associated individual packets in the merged data packet. For the title IDs that are not filtered, exploder 236 divides the associated data packets apart and returns individual packets 238 to router 228 after attaching a header to each of the individual packets 238 that includes a sub- address and title ID. Router 228 then routes each of the individual packets 238 to a device application 239 based on the sub address and title ID.

Detailed Description Text (37):

FIG. 7 shows a block diagram depicting the transformation of data that occurs through the method described in FIG. 6. In FIG. 7, data packets stored in steps 286 and 302 of FIG. 6 are grouped together in memory as message block 310. Each message includes a group identifier, an article identifier, and a data block. The group

identifier identifies the address and sub address of the message and the article identifier provides a title for the message. Message 312 is an example of the messages in message block 310. It includes a group identifier 314 denoted as "News", a title identifier 316 denoted as "Article 1" and a data block 318. Note that each of the messages in message block 310 have the same group identifier denoted as "News" in FIG. 7. Message block 310 is passed through the pre-packaging steps of FIG. 6, denoted as pre-packaging 320 in FIG. 7, to produce a merged packet 322.

Detailed Description Text (38):

Merged packet 322 includes a header 324 and a data segment 326. Header 324 includes a group identifier 328 that identifies the sub-address for each of the data blocks in data segment 326. In the example of FIG. 7, group identifier 328 is "News". Although, group identifier 328 is shown as "News" in FIG. 7, those skilled in the art will recognize that in an actual implementation, group identifier 328 will be a hexadecimal value representing the sub-address.

Detailed Description Text (39):

After group identifier 328, header 324 includes an offset 330 that indicates the location of the first data block 332 in data segment 326. Offset 330 is calculated relative to the last bit of group ID 328. After offset 330, header 324 includes a title identifier and a length for each data block in data segment 326. Thus, first data block 332 has a title identifier 334 and a length 336 that follow offset 330. Second data block 338 has title identifier 340 and length 342 that follow length 336. Similar title identifiers and lengths are provided for each data block in data segment 326. After the last length in header 324, data segment 326 begins and provides data blocks in the same order as their title identifiers and lengths appear in header 324. The data blocks in data segment 326 do not include group identifiers or title identifiers.

Detailed Description Text (41):

Although the present invention has been described in a system that uses address, sub-address and title ID filtering, the invention can also be practiced with systems that only provide address and sub-address filtering or that only provide address filtering. For systems that only provide address filtering, different packets destined for multiple sub-addresses can be included in the same merged data packet.

Current US Original Classification (1):

709/247

Current US Cross Reference Classification (1):

709/206

Current US Cross Reference Classification (2):

709/234

CLAIMS:

5. The computer system of claim 4 wherein the header further comprises a title for each data packet in the merged data packet.

6. The computer system of claim 5 wherein the titles and lengths of the data packets are arranged in the same order as the data packets are arranged in the merged data packet.

16. The method of claim 13 wherein pre-pending a header comprises pre-pending a header that identifies titles for the first and second data packets.



US006304914B1

(12) **United States Patent**  
Deo et al.

(10) **Patent No.:** US 6,304,914 B1  
(45) **Date of Patent:** Oct. 16, 2001

(54) **METHOD AND APPARATUS FOR PRE-COMPRESSION PACKAGING**

(75) **Inventors:** Vinay Deo, Bellevue; Don Kadyk, Bothell; Michael J. O'Leary, Redmond, all of WA (US)

(73) **Assignee:** Microsoft Corporation, Redmond, WA (US)

(\*) **Notice:** Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) **Appl. No.:** 09/158,678

(22) **Filed:** Sep. 22, 1998

(51) **Int. Cl.<sup>7</sup>** ..... G06F 15/16; G06F 13/38; G06F 15/177

(52) **U.S. Cl.** ..... 709/247; 709/234; 709/206

(58) **Field of Search** ..... 709/247, 246, 709/236, 231, 233, 245; 707/101; 341/60; 382/232, 233; 370/447, 521; 708/203

(56) **References Cited**

#### U.S. PATENT DOCUMENTS

5,548,532	*	8/1996	Menand et al.	364/514
5,561,805	*	10/1996	Bruck et al.	395/800
5,579,316	*	11/1996	Venters et al.	370/94.1
5,701,302	*	12/1997	Geiger	370/521
5,754,768	*	5/1998	Brech et al.	709/230
5,774,467	*	6/1998	Herrera Van Der Nood et al.	370/428
5,850,526	*	12/1998	Chou	709/247
5,946,633	*	8/1999	McAlinden	455/551
5,949,418	*	9/1999	Shields et al.	345/342
5,966,385	*	10/1999	Fujii et al.	370/465
6,003,089	*	12/1999	Shaffer et al.	709/233
6,032,197	*	2/2000	Birdwell et al.	709/247
6,052,735	*	4/2000	Ulrich et al.	709/236

6,134,245 \* 10/2000 Scarmalis ..... 370/474  
9,107,899 6/1998 Tuniman et al. .

#### FOREIGN PATENT DOCUMENTS

2311701-A \* 10/1997 (GB) ..... H04L/12/56

#### OTHER PUBLICATIONS

Cobb, Preserving Quality of service Guarantees in Spite of Flow Aggregation, Extending Visual Basic's Comm control, Oct. 1998, Department of Computer Science The University of Texas, 9 pages.\*

IBM Technical Bulletin, DekoVert I Program for Systace Types Hooks, Dec. 1991, IBM, vol. 34, No. 7A, pp. 3.\*

IBM Technical Bulletin, End of Packet Optimization when Cell Routing with Early Drop, May 1997, IBM, vol. 40, No. 5, pp. 2.\*

Baker, Reducing network overhead, Apr. 1994, Mercer Island, vol. 12, No. 4, pp. 4.\*

Murase et al., Mobile radio Packet Data Communications in a TDMA Digital Cellular System, Mar. 1997, IEEE, pp. 1034-1038(5).\*

U.S. application No. 09/107,899, Tuniman et al., pending.

\* cited by examiner

*Primary Examiner*—Mark H. Rinehart

*Assistant Examiner*—Bunjoo Jaroenchonwanit

(74) *Attorney, Agent, or Firm*—Theodore M. Magee; Westman, Champlin & Kelly, P.A.

(57) **ABSTRACT**

A computer readable medium in a computer system has instructions for storing a first and second data packet destined for the same address. The first and second data packets are appended together to produce a merged data packet that is then compressed to produce a compressed data packet. The compressed data packet is then sent to the common address.

18 Claims, 7 Drawing Sheets

